

09.19.08

A

09/18/00
92816 U.S. PTO

PATENT APPLICATION TRANSMITTAL LETTER
(Small Entity)

Docket No.
1122.1.005

TO THE ASSISTANT COMMISSIONER FOR PATENTS

Transmitted herewith for filing under 35 U.S.C. 111 and 37 C.F.R. 1.53 is the patent application of:

Jeffrey M. Drew

For: **METHOD AND APPARATUS FOR MINIMIZING FILE SCANNING BY ANTI-VIRUS PROGRAMS**

92816 U.S. PTO
09/664919

09/18/00

Enclosed are:

- ☒ Certificate of Mailing with Express Mail Mailing Label No. EL33485238 US
- ☒ 4 sheets of drawings.
- ☒ A certified copy of a **Utility Patent** application.
- ☒ Declaration ☒ Signed. ☐ Unsigned.
- ☒ Power of Attorney
- ☒ Information Disclosure Statement
- ☐ Preliminary Amendment
- ☐ Verified Statement(s) to Establish Small Entity Status Under 37 C.F.R. 1.9 and 1.27.
- ☒ Other: **Assignment w/Recordation Form, and Check for \$40.00 recording fee.**

CLAIMS AS FILED

For	#Filed	#Allowed	#Extra	Rate	Fee
Total Claims	14	- 20 =	0	x \$18.00	\$0.00
Indep. Claims	5	- 3 =	2	x \$78.00	\$156.00
Multiple Dependent Claims (check if applicable) <input type="checkbox"/>					\$0.00
BASIC FEE					\$690.00
TOTAL FILING FEE					\$846.00

- ☒ A check in the amount of **\$846.00** to cover the filing fee is enclosed.
- ☒ The Commissioner is hereby authorized to charge and credit Deposit Account No. **23-0510** as described below. A duplicate copy of this sheet is enclosed.
 - ☐ Charge the amount of _____ as filing fee.
 - ☒ Credit any overpayment.
 - ☒ Charge any additional filing fees required under 37 C.F.R. 1.16 and 1.17.
 - ☐ Charge the issue fee set in 37 C.F.R. 1.18 at the mailing of the Notice of Allowance, pursuant to 37 C.F.R. 1.311(b).

Dated:

Kenneth Watov
 Signature
 Kenneth Watov, Esquire
 Watov & Kipnes, P.C.
 P.O. Box 247
 Princeton Junction, NJ 08550

CC:

CERTIFICATE OF MAILING BY "EXPRESS MAIL" (37 CFR 1.10)Applicant(s): **Jeffrey M. Drew**

Docket No.

1122.1.005

Serial No.
(Unknown)Filing Date
(Herewith)Examiner
(Unknown)Group Art Unit
(Unknown)1c784 U.S. PTO
09/664919

Invention:

METHOD AND APPARATUS FOR MINIMIZING FILE SCANNING BY ANTI-VIRUS PROGRAMS

I hereby certify that the following correspondence:

Patent Application with associated executed documents, as identified in "Patent Application Transmittal Letter."

(Identify type of correspondence)

is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 CFR 1.10 in an envelope addressed to: The Assistant Commissioner for Patents, Washington, D.C. 20231 on

September 18, 2000
(Date)**Mandy Willever***(Typed or Printed Name of Person Mailing Correspondence)*Mandy Willever*(Signature of Person Mailing Correspondence)***EL 334858238 US***("Express Mail" Mailing Label Number)*

Note: Each paper must have its own certificate of mailing.

EXPRESS MAIL CERTIFICATE	
DATE	September 18, 2000
LABEL NO.	EL 334858238 US
I HEREBY CERTIFY THAT, ON THE DATE INDICATED ABOVE, I DEPOSITED THIS PAPER OR FEE WITH THE U.S. POSTAL SERVICE AND THAT IT WAS ADDRESSED FOR DELIVERY TO THE COMMISSIONER OF PATENTS & TRADEMARKS, WASHINGTON, DC 20231 BY "EXPRESS MAIL POST OFFICE TO ADDRESSEE" SERVICE.	
NAME (PRINT)	Mandy Willever
SIGNATURE	<i>Mandy Willever</i>

METHOD AND APPARATUS FOR MINIMIZING FILE SCANNING BY ANTI-VIRUS PROGRAMS

Field Of The Invention

The present invention relates generally to computer programs for providing anti-virus protection for computers from computer viruses, and more specifically to optimization of the performance of such anti-virus computer programs.

Background Of The Invention

The design and implementation of anti-virus computer programs for protecting computers from damage and/or interruption of operation due to computer viruses are well known in the art. A great deal of time and effort is spent in the design of such anti-virus programs to reduce the amount of computer time required by such programs for detecting viruses in computer files, and preventing detected virus contaminated or infected files from being written onto the hard drive or other storage media associated with a particular computer system or server. The use of computer networks for interconnecting a plurality of computers, either on a local or wide area network, has provided increasingly greater opportunity for so-

5 called "computer hackers" to introduce viruses into the associated computers. The now
widespread use of the Internet, and World Wide Web, has caused a major increase in the
introduction of computer viruses into computer systems connected to such networks. In turn,
anti-virus programs have to be continuously updated and expanded in order to recognize,
cope with, and cleanse infected computer files of myriad viruses that may be introduced by
10 hackers. As the number of computer viruses scanned for by anti-virus programs increases,
the time required for scanning a given file for any such viruses increases in proportion to the
increase in the number of viruses. Accordingly, computer programmers associated with the
design of anti-virus computer programs are continuously searching for methods to reduce the
computer time these programs must spend in scanning files for viruses.

15 In U.S. 5,649,095, entitled "Method and Apparatus For Detecting Computer Viruses
Through The Use Of A Scan Information Cache", the length information of one portion of a
file, e.g. a fork, is upon opening stored in a cache. Upon initiating a scan of the file, the
length of the portion of the file corresponding to the portion in cache is compared to the
length of the latter. If a size difference is detected, the file is only scanned for viruses which
20 cause a change in the length or size of that portion of a file, thereby eliminating spending time
scanning for other viruses. The teachings of U.S. 5,649,095 are incorporated herein by
reference to the extent they do not conflict herewith.

In U.S. Serial No. 09/481,060, filed January 11, 2000, owned by the same Assignee
as the present invention, and entitled "Fast Virus Scanning Using Scanning Stamping," a

5 unique session key is created for each execution of anti-virus software, and is used to create a session stamp for each file scanned during the associated execution. The session stamps are stored for use by the anti-virus software to validate a session stamp when a request for an associated file is made. A file is scanned if the session stamp is invalid or absent for that file.

Summary Of The Invention

10 An object of the present invention is to provide a method for minimizing the scanning of an opened file for viruses between the time a user requests closure of the file, and the time that the file is actually closed, typically upon writing back to a hard disk or to a floppy disk.

15 In one embodiment of the invention, the aforesaid object and other objects of the invention are met by including computer code in the anti-virus program to respond to a request for closure of a file by a user of a particular computer or computer system, by first determining whether a flag has been set or raised by the operating system indicating that the file was modified between the time it was open to the time when the user requested closure of the file. Certain operating systems provide such modification flags through the use of a "dirty cache buffer." If such a flag is set or raised indicating that the file was modified during this
20 time, the computer coding causes the anti-virus program to scan the file for known viruses, and if a file is found to be infected by a virus, the anti-virus program prevents the file from being written back into the hard disk, or other storage media of the system, until such time that the file is cleansed of the virus. However, if no flag was detected indicating that the file

5 was modified during the time that it was opened, then the file is considered to be unmodified and free of viruses. The operating system is then released by the antivirus program to write the file into the desired storage media, such as a hard disk. Accordingly, the present invention avoids wasting valuable computer time in scanning open files, or checking file information caches, for viruses before they are closed, by taking advantage of operating systems that are designed to raise or set a flag whenever a file that has been opened is modified during the time that it has been opened.

10 In a second embodiment of the invention, if upon a user request to close an open file a modification flag is detected for the file, the next step is to determine whether the file was modified in a portion of the file that viruses can enter. If such a modification was made, the file is scanned. If such a modification was not made, the operating system is released to close the file.

Brief Description Of The Drawings

15 Various embodiments of the present invention will be described in detail below with reference to the drawings, in which like items are identified by the same reference designation, wherein:

20 Figure 1 shows a typical local area computer network of the prior art, in this example Ethernet, for connecting a network server network operating system with a plurality of personal computer operating systems, such as desktop systems, laptop computers, and so

5
forth;

Figure 2 is a flowchart showing the traditional scanning method of the prior art for the typical steps associated with scanning an open file for viruses;

Figure 3 shows a flowchart for one embodiment of the invention for minimizing the unnecessary scanning of an open file for viruses before the file is closed; and

10
Figure 4 shows another embodiment of the invention for minimizing the necessity of scanning an open file for viruses before it is closed in the presence of a file information cache or any other optimization cache.

Detailed Description Of The Invention

As shown in Figure 1, a typical local area network known as an Ethernet 2 is used to
15 permit a network server computer 4 loaded with an appropriate network operating system 6 to communicate with a plurality of personal computers such as desk top personal computers 8, and typical laptop personal computers 10, include a hard disk drive 16 for storing a PC operating system 12, and other programs and data. Disk buffers 14 provide an interface between the hard disk drive 16 and the central processing unit (not shown) of the personal
20 computer 8, for permitting the operating system 12 to provide computer code for running the

central processing unit, and other subsystems of the personal computer 8. A similar configuration is used in laptop 10. In the server computer 4, a memory device 18 is provided for storing files, and for storing an operating system for driving the central processing unit (not shown) of the server 4, in applications where another memory device is not available for storing the network operating system 6. Cache buffers 20 provide an interface for the temporary storage of files retrieved from the file storage memory 18 for distribution from the server to 1 of the personal computers 8 or laptop computers 10, in this example, connected to the Ethernet 2.

In the present state of the art, computer systems that are properly configured for providing protection against computer viruses, typically permit the opening and closing of files using the steps shown in the flowchart of Figure 2. A user of a personal computer 8 or laptop computer 10, as shown in Figure 1, can request the server computer 4 to open a file for write access, as indicated by step 22. Please note that although various embodiments of the present invention are described in association with a computer network, such as the Ethernet 2 of Figure 1, the invention is not so limited, and can be implemented through use of any other known network. Also, various embodiments of the present invention are applicable for use by a user directly on their own dedicated personal computer 8 or laptop 10. For these and other computer configurations, it is typical after a file is opened for write access in step 22, to next scan the file for viruses in step 24, via a computer anti-virus program. If no viruses are detected in step 24, step 26 is entered for providing the requested file to the Application program of the user. A period of time after the file is opened, it is typical that a user will

5 request that the file be closed. Upon a file closure request being made in step 28, the typical anti-virus program loaded into a computer system, such as network server computer 4, interfaces with the network operating system 6 to scan the file for viruses in step 30 before permitting the file to be written back into memory. As shown in decision step 32, if a file is found to be infected with a virus, the anti-virus program proceeds to step 34 for preventing
10 the infected file from being written into memory, such as file storage memory 18.

Alternatively, if in step 32 no virus is uncovered, the anti-virus program proceeds to step 36 for allowing the operating system to write the file back into memory. The last step 38, indicative that scanning has been completed, terminates the typical virus protection program scanning routine. Note that in some state-of-art operating systems, the cache buffers 20 are used to store files upon opening in an unmodified state. Before step 36, the file to be closed
15 is compared to the corresponding unmodified file in a cache buffer memory 30. If the file to be closed is found to be identical to the unmodified cached file, write step 36 is skipped, and the open file is closed with only the file's time stamp being updated. In computers not loaded with an anti-virus program, the file comparison occurs after step 28.

20 With reference to Figure 3, a first embodiment of the invention provides additional steps for an anti-virus program operating in conjunction with an operating system. Note that steps 22, 24, 26, 28, 30, 32, 34, 36, and 38 are the same as the traditional scanning steps for an anti-virus program of Figure 2. In this first embodiment of the invention, new steps 40 and 42 are included. As shown, step 40 is a decision step interposed between steps 28 and
25 30. Decision step 40 determines whether a file was actually written, that is modified by the

5 user performing some writing step on the open file. The computer coding for step 40 determines whether an open file was actually written or modified by looking for a flag in the operating system indicative of such modification. Many state-of-art operating systems provide such flags, which are used by the operating system to avoid rewriting a file back into memory if it has not been modified. If the anti-virus program in step 40 does find that a
10 modification flag has been set or raised by the operating system, then step 30 is entered into for scanning the file for viruses. If on the other hand no modification flag is found, the file is considered clean, and the operating system is allowed to write the file back into the memory in step 42, as shown, or simply close the file. If an operating system does not provide such modification flags, the first embodiment of the invention can be extended to add cache buffer memories 20, and appropriate computer coding to incorporate the modification flag function
15 into the operating system.

Note that one operating system that provides modification flags is the Novell NetWare 4.X® and later versions, which has a function call FEGetOpenFileInfo(). One of the parameters of this function is a file handle that indicates a file has been opened. The
20 aforesaid function call also includes a parameter known as "flags" field for providing flags to indicate the status of the file. Typically, such flags are not well documented by Novell®, but the inventor has determined through inquiry that such flags can be used to obtain what is known as a "dirty cache buffer" state of the file. Such a flag provides an indication of whether a file was modified. The associated operating system uses such flags for optimizing
25 closure of open files, by avoiding the time for rewriting to disk if the file was not modified as

5 indicated by the lack of the modification flag being set or raised. As previously indicated, the first embodiment of the present invention uses the absence of such modification flags after a call for closure of a file to avoid scanning the file for viruses, and uses the presence of such a modification flag to scan the file for viruses before permitting writing of the file back into memory.

10 A second embodiment is shown in the flowchart of Figure 4. In comparing this second embodiment of the invention with the first embodiment shown in Figure 3, note that the second embodiment includes a new step 44 between both steps 24 and 26, and another new step 46 between both steps 40 and 42, and steps 40 and 30, as shown in Figure 4. More specifically, step 44 uses a full cache buffer memory 20 for storing an entire file. Typically there are specific portions of a file that a virus must use by necessity in order to invade the file. In operation of the programming steps of the second embodiment of the invention, if in step 40 a flag indicative of modification of the file is not detected, the present computer program proceeds through steps 42 and 38, as in the first embodiment of the invention of Figure 3. If however, a modification flag is detected, step 46 is entered for determining whether the file was modified in a manner that would permit a virus to invade the file. Step 46 is carried out by determining whether any of the file portions stored in step 44, when compared to the open file for which a closure has been requested, has changed, indicating a file modification. If not modified, the programming proceeds with steps 42 and 48. If however, it is determined that the file was modified in a manner to permit a virus to invade the file, such as the head end being changed, or a macro being changed or added in the word

15
20
25

5 portion, the anti-virus program proceeds to step 30, and therefrom to step 32 and operates as previously described for the first embodiment of the invention of Figure 3.

Note that there are anti-virus programs known in the art that utilize a cache memory for storing data or file information upon the opening of a particular file, which information is indicative of the unmodified section or computer coding of a file that must be modified in order for a virus of a particular type to have an opportunity to invade that file. A cache memory is used for storing such file coding for every virus the anti-virus program is capable of scanning for to prevent entry into the protected computer. Cozza U.S. Patent No. 5,649,095 teaches the use of such a scan information cache for detecting a plurality of computer viruses, whereby a "fork" portion of a file is stored when the file is opened. If a request for closure of the particular file is made, the stored file data is compared to the same file data of the open file now requested for closure to determine if that data has been modified since opening the file. If modified, the anti-virus program scans the file for the type of virus that would invade that type of data or fork code portion of the file. However, the present inventor does not know of any anti-virus programs that combine a file data storage step, such as step 44, in combination with step 40 to determine whether an operating system has raised a modification flag, for providing criteria, such as in steps 40 and 46, for causing the anti-virus program to proceed to scan a file for viruses, as in step 30. Nor does the present inventor know of any anti-virus programs that monitor an operating system for the raising of a flag that indicates a file has been modified since opening, for example by accessing a preexisting "dirty cache buffer" in a server's operating system to check for the flag, for triggering the

5 scanning of the file for viruses, or for avoiding scanning of a file for viruses if no flag has been raised, as in the first embodiment of the invention. Contrary to the teaching of 5,649,095 for storing a "fork" portion of a file, as indicated above, the second embodiment of the invention creates a "dirty cache buffer" in step 44 for storing the entire file for monitoring.

10 Although various embodiments of the invention have been shown and described, they are not meant to be limiting. Those of ordinary skill in the art may recognize certain modifications to these embodiments, which modifications are meant to be covered by the spirit and scope of the appended claims. For example, in another embodiment of the invention, network protocols are monitored to determine if a write packet was received by an associated computer or file server for a given open file to detect that a write event has occurred.

15

What Is Claimed Is:

1. A method for optimizing the operation of an anti-virus computer program for use with an operating system, comprising the steps of:
 - detecting a request for closure of an opened computer file;
 - determining in response to a closure request if the opened computer file has been modified since being opened;
 - scanning said opened file for viruses before closure only if said opened file has been modified; and
 - closing said file if unmodified, and closing said file after scanning for viruses if found virus free.
2. The method of Claim 1, further including before said detecting step, the steps of:
 - determining whether said operating system includes a "dirty cache buffer" to raise or set a modification flag relative to a file being modified during the time it has been open, a computer code being indicative of said flag; and
 - using the computer code for a raised or set modification flag, if available, for carrying out said modification determining step by checking for the presence of a raised modification for said file.

1 3. The method of Claim 2, wherein if it is determined that said operating system
2 does not provide a file modification flag, said method further includes the steps of:

3 establishing a "dirty cache buffer";

4 and

5 raising a modification flag in said "dirty cache buffer" if an opened file associated
6 with said flag has been modified by a write operation.

1 4. The method of Claim 1, wherein said operating system includes a "dirty cache
2 buffer" for providing a computer code for a modification flag indicative of the modification
3 of an open file, said method further including in said modification determining step, the step
4 of:

5 detecting the presence of said modification flag to determine if the associated
6 opened file has been modified.

1 5. The method of claim 4, further including the steps of:
2 scanning a file for viruses in response to a request for opening the file;
3 opening said file if virus free;
4 establishing a cache buffer memory for storing upon opening of a file only a virus
5 vulnerable portion of that file that a virus must use to enter and infect said file;
6 said modification determining step including the steps of:
7 indicating an open file is unmodified in the absence of an associated
8 modification flag;

9 responding to the presence of a modification flag by comparing a portion of
10 said open file to the associated unmodified virus vulnerable portion of said file in said cache
11 buffer memory to determine if the portion of the open file has been modified since the
12 opening of the file;
13 indicating the opened file is unmodified is the virus vulnerable portion is
14 unmodified; and
15 indicating the opened file is modified if the virus vulnerable portion is
16 modified.

1 6. The method of Claim 1, wherein said step of determining in response to a closing
2 request if the opened computer file has been modified since being opened includes the step
3 of:
4 monitoring network protocols to determining if a write packet was initiated for a
5 given open file.

1 7. A method for optimizing operation of an anti-virus program in an operating
2 system, said operating system including programming for raising a flag indicative of
3 modification of an open file during the time the file has been open, said method including the
4 steps of:
5 detecting the event of a request for closing said file being made to said
6 operating system;
7 determining whether said modification flag has been raised by said operating

8 system for said open file;

9 scanning said open file, in response to said modification flag, for viruses

10 before permitting said operating system to close said file; and

11 skipping said step of scanning for viruses before closure of said open file,

12 whenever said modification flag is not present.

1 8. A method for optimizing the operation of an anti-virus program in use in an
2 operating system, said operating system including programming for raising a flag indicative
3 of modification of an open file during the time the file has been open, said method including
4 the steps of:

5 scanning a file for viruses in response to a request from an associated
6 computer user to open and gain access to said file;

7 permitting said file to be opened if virus free;

8 storing upon opening a virus vulnerable unmodified portion of said open file;

9 detecting the event of a request for closing said open file being made to said
10 operating system;

11 determining whether said modification flag has been raised by said operating
12 system for said open file;

13 scanning said open file, in response to said modification flag, for viruses
14 before permitting said operating system to close said file; and

15 skipping said step of scanning for viruses before closure of said open file,
16 whenever said modification flag is not present;

17 responding to the presence of a modification flag by comparing the stored
18 unmodified virus vulnerable portion of said file to the associated portion of said open file to
19 determine if that portion has been modified during the time the file has been open; and
20 skipping said step of scanning for viruses before closure of said open file if the
21 virus vulnerable portion of said open file is unmodified.

1 9. A computer program product for detecting computer viruses on a file server,
2 the file server providing file storage and retrieval services for at least one client computer
3 over a network, said computer program product comprising:

4 computer code for detecting an open request from a client computer, the open request
5 asking for a requested file from the file server;

6 computer code for scanning said requested file for computer viruses, whereby the file
7 server is permitted to provide said requested file to the client computer if no computer viruses
8 are found therein;

9 computer code for detecting a close request from the client computer associated with
10 said requested file;

11 computer code for accessing an operating system flag that indicates whether the
12 requested file was changed prior to said close request;

13 computer code for scanning said requested file for computer viruses if said requested
14 file was changed prior to said close request; and

15 computer code for skipping scanning said requested file if it was not changed prior to
16 said close request.

1 10. The computer program product of claim 9, wherein said operating system flag
2 is generated externally to said computer program product by the operating system in order to
3 reduce redundant disk writes, whereby said computer code for scanning is invoked upon
4 closing of the requested file only when actual disk writes are made by the operating system
5 for the requested file.

1 11. The computer program product of claim 10, wherein said computer code for
2 accessing uses a file handle generated by the operating system to identify the operating
3 system flag corresponding to the requested file, said handle having been generated when the
4 file was opened.

1 12. A computer program product for detecting computer viruses on a file server,
2 the file server providing file storage and retrieval services for at least one client computer
3 over a network, said computer program product comprising:

4 computer code for detecting an open request from a client computer, the open request
5 asking for a requested file from the file server;

6 computer code for scanning said requested file for computer viruses, whereby the file
7 server is permitted to provide said requested file to the client computer if no computer viruses
8 are found therein;

9 computer code for detecting a close request from the client computer associated with
10 said requested file;

11 computer code for accessing an operating system flag that indicates whether the
12 requested file was changed prior to said close request;
13 computer code for skipping scanning said requested file if it was not changed prior to
14 said close request;
15 computer code responsive to said requested file having been changed prior to said
16 close request for determining whether a virus vulnerable portion of said file was changed;
17 computer code for skipping scanning said requested file if a virus vulnerable portion
18 of said file was not changed prior to said close request; and
19 computer code for scanning said requested file if a virus vulnerable portion of said file
20 was changed prior to said close request.

21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
221

ABSTRACT OF THE INVENTION

Scanning time for a computer anti-virus program is minimized by eliminating scanning of a file for viruses before closure, in response to the absence of a modification flag being raised in an associated operating system, the flag being indicative of the file having been modified between the time the file was opened to the time of a close request.

1 OF 4

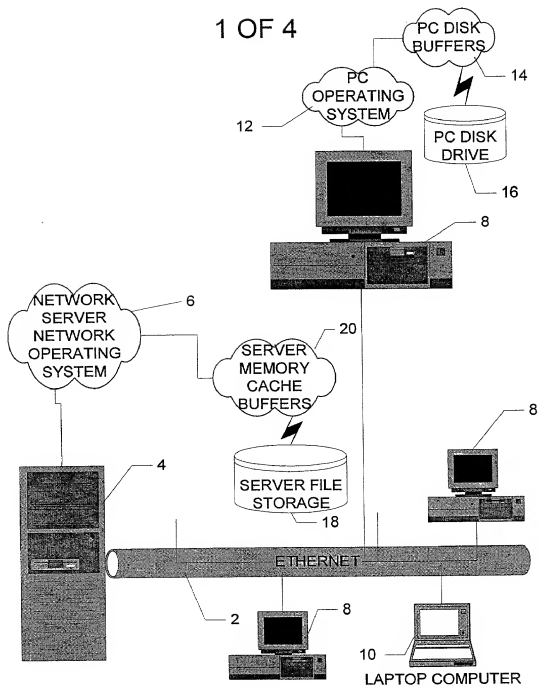


FIG. 1 (PRIOR ART)

2 OF 4

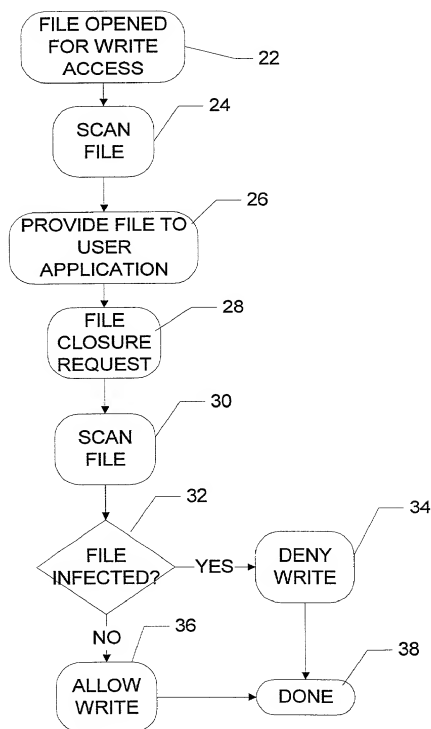


FIG. 2 (PRIOR ART)

3 OF 4

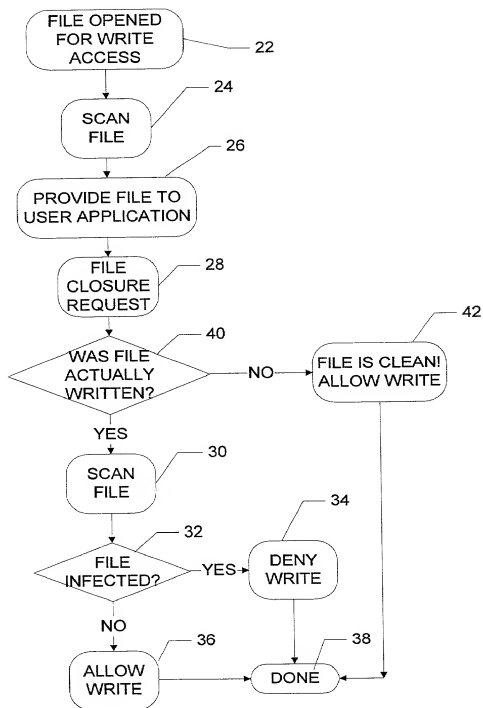


FIG. 3

4 OF 4

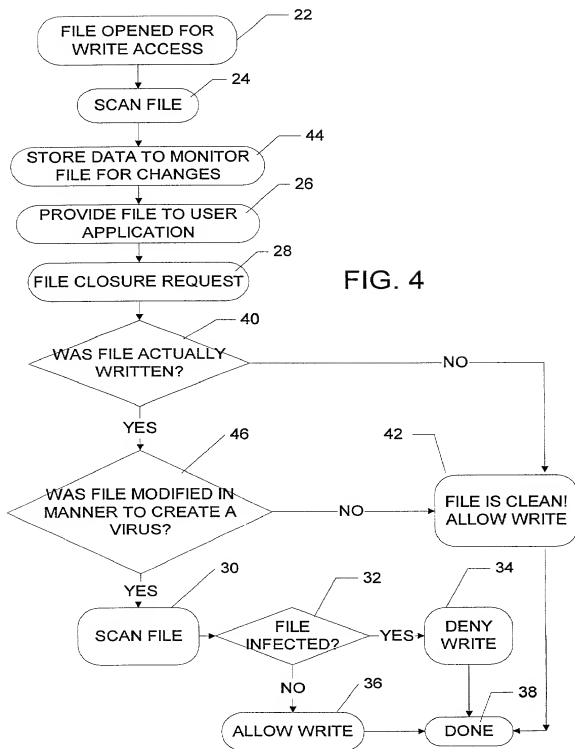


FIG. 4

Docket No.
1122.1.005

Declaration and Power of Attorney For Patent Application

English Language Declaration

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name,

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled
METHOD AND APPARATUS FOR MINIMIZING FILE SCANNING BY ANTI-VIRUS PROGRAMS

the specification of which

(check one)

☒ is attached hereto.

☐ was filed on _____ as United States Application No. or PCT International

Application Number _____

and was amended on _____

(if applicable)

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

I acknowledge the duty to disclose to the United States Patent and Trademark Office all information known to me to be material to patentability as defined in Title 37, Code of Federal Regulations, Section 1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, Section 119(a)-(d) or Section 365(b) of any foreign application(s) for patent or inventor's certificate, or Section 365(a) of any PCT International application which designated at least one country other than the United States, listed below and have also identified below, by checking the box, any foreign application for patent or inventor's certificate or PCT International application having a filing date before that of the application on which priority is claimed.

Prior Foreign Application(s)

Priority Not Claimed

(Number)

(Country)

(Day/Month/Year Filed)

☐

(Number)

(Country)

(Day/Month/Year Filed)

☐

(Number)

(Country)

(Day/Month/Year Filed)

☐

I hereby claim the benefit under 35 U.S.C. Section 119(e) of any United States provisional application(s) listed below:

(Application Serial No.)

(Filing Date)

(Application Serial No.)

(Filing Date)

(Application Serial No.)

(Filing Date)

I hereby claim the benefit under 35 U. S. C. Section 120 of any United States application(s), or Section 365(c) of any PCT International application designating the United States, listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States or PCT International application in the manner provided by the first paragraph of 35 U.S.C. Section 112, I acknowledge the duty to disclose to the United States Patent and Trademark Office all information known to me to be material to patentability as defined in Title 37, C. F. R., Section 1.56 which became available between the filing date of the prior application and the national or PCT International filing date of this application:

(Application Serial No.)

(Filing Date)

(Status)
(patented, pending, abandoned)

(Application Serial No.)

(Filing Date)

(Status)
(patented, pending, abandoned)

(Application Serial No.)

(Filing Date)

(Status)
(patented, pending, abandoned)


I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith. *(list name and registration number)*

Kenneth Watov, Esquire
Registration No. 26,042

Send Correspondence to: **Kenneth Watov, Esquire**
Watov & Kipnes, P.C.
P.O. Box 247
Princeton Junction, NJ 08550

Direct Telephone Calls to: *(name and telephone number)*
Kenneth Watov, Esquire (609) 243-0330

Full name of sole or first inventor	
Jeffrey M. Drew	
Sole or first inventor's signature	Date
	9/15/2000
Residence	
140 Deepdale Drive, Middletown, NJ 07748	
Citizenship	
United States of America	
Post Office Address	
140 Deepdale Drive, Middletown, NJ 07748	

Full name of second inventor, if any	
Second inventor's signature	Date
Residence	
Citizenship	
Post Office Address	